AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method of performing a recursion process on a data block for error correction, comprising:

concurrently operating pipelined sub-processes of a recursion process on the data block, wherein the pipelined sub-processes perform error correction on the data block and are implemented as sub-circuits in an integrated circuit;

storing output data from each sub-process; and

inputting the output data of each sub-process to a subsequent sub-process of the pipelined sub-processes[[.]],

wherein concurrently operating the pipelined sub-processes includes:

performing a partial calculation by each sub-process, of a probability value output by the recursion process for each sample within the data block; and

performing by a first sub-process, a partial calculation of the probability value for a first sample in a first cycle, performing by a second sub-process, a partial calculation of the probability value for the first sample in a second cycle that occurs after the first cycle, and performing by the first sub-process, a partial calculation of the probability value for a second sample in the second cycle.

2. (Original) The method of claim 1, wherein concurrently operating the pipelined sub-processes comprises concurrently performing a particular recursion operation of the recursion process on a plurality of samples of the data block.

3. (Original) The method of claim 1, wherein concurrently operating the pipelined sub-processes comprises concurrently performing a plurality of recursion operations of the recursion process on a particular sample of the data block.

4. (Original) The method of claim 1, wherein concurrently operating the pipelined sub-processes comprises:

concurrently performing a particular recursion operation of the recursion process on a plurality of samples of the data block; and

concurrently performing a plurality of recursion operations of the recursion process on a particular sample of the data block.

5. (Currently Amended) A method of performing a forward/backward recursion process for error correction in decoding an encoded data block, comprising:

dividing the encoded data block into data windows;

decoding the data windows with error correction using concurrently operating pipelined sub-processes of the forward/backward recursion process, the pipelined sub-processes implemented as respective sub-circuits of an integrated circuit; and

storing an output value from each sub-process for input to a subsequent sub-process of the pipelined sub-processes[[.],

wherein decoding the data windows includes:

performing a partial calculation by each sub-process, of a an alpha/beta probability value output by the recursion process for each sample within the window; and

performing by a first sub-process, a partial calculation of the alpha/beta probability value for a first sample in a first cycle, performing by a second sub-process, a partial calculation of the alpha/beta probability value for the first sample in a second cycle that occurs after the first cycle, and performing by the first sub-process, a partial calculation of the alpha/beta probability value for a second sample in the second cycle.

6. (Original) The method of claim 5, wherein decoding the data windows comprises concurrently performing a particular recursion operation of the forward/backward recursion process on a plurality of the data windows.

7. (Original) The method of claim 6, wherein concurrently performing the particular recursion operation on the plurality of data windows comprises concurrently performing one of an alpha recursion and a beta recursion on the plurality of the data windows.

8. (Original) The method of claim 6, wherein concurrently performing the particular recursion operation on the plurality of data windows comprises performing one of a forward recursion and a backward recursion on the plurality of data windows.

9. (Original) The method of claim 6, wherein concurrently performing the particular recursion operation on the plurality data windows comprises executing a particular sub-process to perform one of a forward recursion and a backward recursion on a particular data window during alternate processor cycles.

10. (Original) The method of claim 5, wherein concurrently operating the pipelined sub-processes comprises concurrently performing a plurality of recursion operations of the forward/backward recursion process on a particular data window.

11. (Original) The method of claim 10, wherein concurrently performing the plurality of recursion operations comprises executing a particular sub-process to perform one of a forward recursion and a backward recursion on the particular data window during alternate processor cycles.

12. (Original) The method of claim 10, wherein concurrently performing the plurality of recursion operations comprises concurrently performing an alpha recursion and a beta recursion.

13. (Original) The method of claim 1, wherein concurrently operating the pipelined sub-processes comprises concurrently performing a particular recursion operation of the forward/backward recursion process on a plurality of the data windows and a plurality of recursion operations of the forward/backward recursion process on a particular data window.

14. (Original) The method of claim 1, wherein storing the output of each sub-process comprises storing the output using a semiconductor memory.

15. (Original) The method of claim 1, wherein storing the output of each sub-process comprises storing the output using a delay element.

16. (Currently Amended) A data decoder sub-system, comprising:

an input circuit configured to receive an encoded data block; and

one or more processors coupled to the input circuit, the one or more processors implemented in an integrated circuit and configured to concurrently operate pipelined sup-processes of a recursion process that decode the data block with error correction, the one or more processors further configured to store an output of each sub-process for input to a subsequent sub-process of the pipelined sub-processes[[.]],

wherein each sub-process performs a partial calculation of a probability value output by the recursion process for each sample within the data block, and

wherein a first sub-process performs a partial calculation of the probability value for a first sample in a first cycle, a second sub-process performs a partial calculation of the probability value for the first sample in a second cycle that occurs after the first cycle, and the first sub-process performs a partial calculation of the probability value for a second sample in the second cycle.

17. (Original) The sub-system of claim 16, wherein the pipelined sub-processes concurrently perform a particular recursion operation of the recursion process on a plurality of samples of the data block.

18. (Original) The sub-system of claim 16, wherein the pipelined sub-processes concurrently perform a plurality of recursion operations of the recursion process on a particular sample of the data block.

19. (Original) The sub-system of claim 16, wherein the recursion process comprises at least one of a forward recursion operation and a backward recursion operation.

20. (Original) The subsystem of claim 16, wherein:

the input circuit is configured to divide the data block into data windows; and

the pipelined sub-processes are concurrently operated to perform the recursion process in decoding the data windows.

21. (Original) The sub-system of claim 16, wherein:

the recursion process comprises an alpha/beta recursion process; and

the alpha/beta recursion process comprises an alpha recursion operation and a beta recursion operation.

22. (Original) The sub-system of claim 16, wherein the recursion process comprises

a forward/backward recursion process; and

the forward/backward recursion process comprises a forward recursion operation and a backward recursion operation.

23. (Original) The sub-system of claim 16, wherein the pipelined sub-processes are concurrently operated to perform a particular recursion operation of the recursion process on a plurality of the data windows.

24. (Original) The sub-system of claim 23, wherein the particular recursion operation is one of a forward recursion and a backward recursion.

25. (Original) The sub-system of claim 23, wherein the particular recursion operation is one of an alpha recursion and a beta recursion.

26. (Original) The sub-system of claim 16, wherein the pipelined sub-processes are concurrently operated to perform a plurality of recursion operations of the recursion process on a particular data window.

27. (Original) The sub-system of claim 16, wherein the pipelined sub-processes concurrently perform a particular recursion operation of the recursion process on a plurality of the data windows and a plurality of recursion operations of the recursion process on a particular data window.

28. (Original) The sub-system of claim 16, wherein each of the one or more processors comprise a memory element that stores the output of each sub-process.

29. (Original) The sub-system of claim 16, wherein each of the one or more processors comprise a delay element that stores the output of each sub-process.

30. (Currently Amended) A turbo decoder system, comprising:

an input circuit configured to receive a data block; and

one or more soft-input-soft-output decoders coupled to the input circuit and configured to decode the data block with error correction, the one or more soft-input-soft-output decoders implemented in an integrated circuit, and at least one of the soft-input-soft-output decoders including,

a gamma calculator configured to calculate a state transition probability produced by a particular input bit of the data block;

an alpha/beta recursion processor coupled to the gamma calculator and configured to perform an alpha/beta recursion process, the alpha/beta recursion processor configured to concurrently operate pipelined sub-processes that perform the alpha/beta recursion process and to store the output of each sub-process for input by a subsequent sub-process of the pipelined sub-processes; [[and]]

wherein each sub-process performs a partial calculation of a probability value output by the recursion process for each sample within the data block;

wherein a first sub-process performs a partial calculation of the probability value for a first sample in a first cycle, a second sub-process performs a partial calculation of the probability value for the first sample in a second cycle that occurs after the first cycle, and the first sub-process performs a partial calculation of the probability value for a second sample in the second cycle; and

a log-likelihood-ratio processor coupled to the gamma calculator and the alpha/beta recursion processor and configured to provide an estimate of a transmitted data block.

31. (Original) The turbo decoder system of claim 30, wherein the pipelined sub-processes concurrently perform a particular recursion operation of the alpha/beta recursion process on a plurality of samples of the data block.

32. (Original) The turbo decoder system of claim 30, wherein the pipelined sub-processes concurrently perform a plurality of recursion operations of the alpha/beta recursion process on a particular sample of the data block.

33. (Original) The turbo decoder system of claim 30, wherein:

the input circuit is configured to divide the data block into data windows; and

the pipelined sub-processes are concurrently operated to perform the alpha/beta recursion process in decoding the data windows.

34. (Original) The turbo decoder system of claim 33, wherein the pipelined sub-processes are concurrently operated to perform a particular recursion operation of the recursion process on a plurality of the data windows.

35. (Original) The sub-system of claim 33, wherein the pipelined sub-processes are concurrently operated to perform a plurality of recursion operations of the recursion process on a particular data window.

36. (Original) The sub-system of claim 33, wherein the pipelined sub-processes concurrently perform a particular recursion operation of the recursion process on a plurality of the data windows and a plurality of recursion operations of the recursion process on a particular data window.

37. (Currently Amended) A system for performing a recursion process on a data block for error correction, comprising:

means for concurrently operating pipelined sub-processes of a recursion process on the data block, wherein the pipelined sub-processes perform error correction on the data block and are implemented as respective sub-circuits of an integrated circuit;

means for storing output data from of each sub-process; and

means for inputting the output data of each sub-process to a subsequent sub-process of the pipelined sub-processes[[.]],

wherein each sub-circuit that implements a sub-process performs a partial calculation of an alpha/beta probability of the forward/backward recursion process for each sample within the data block, and

wherein a first sub-process performs a partial calculation of the alpha/beta probability value for a first sample in a first cycle, a second sub-process performs a partial calculation of the alpha/beta probability value for the first sample in a second cycle that occurs after the first cycle, and the first sub-process performs a partial calculation of the alpha/beta probability value for a second sample in the second cycle.

38. (Currently Amended) A system for performing a forward/backward recursion process for error correction in decoding an encoded data block, comprising:

means for dividing the encoded data block into data windows;

means for decoding the data windows with error correction using concurrently operating pipelined sub-processes of the forward/backward recursion process, the pipelined sub-processes implemented as respective sub-circuits of an integrated circuit; and

means for storing output data from each sub-process for input by a subsequent sub-process of the pipelined sub-processes[[.]],

wherein each sub-circuit that implements a sub-process performs a partial calculation of an alpha/beta probability value of the forward/backward recursion process for each sample within the window, and

wherein a first sub-process performs a partial calculation of the alpha/beta probability value for a first sample in a first cycle, a second sub-process performs a partial calculation of the alpha/beta probability value for the first sample in a second cycle that occurs after the first cycle, and the first sub-process performs a partial calculation of the alpha/beta probability value for a second sample in the second cycle.